

TD1 : Organisation et stockage des données

Exerice 1 :

On dispose d'une base de 3 Go constituée d'enregistrements dont la taille moyenne est 3 000 octets.

1. Combien de temps prend la lecture complète de cette base avec un parcours séquentiel ?
2. Combien de temps prend la lecture en effectuant une lecture physique aléatoire pour chaque enregistrement ?

Voir le tableau des caractéristiques des mémoires du support de cours.

Exerice 2 :

Le tableau ci-dessous donne les spécifications partielles d'un disque magnétique. Répondez aux questions suivantes.

1. Quelle est la capacité moyenne d'une piste?, d'un cylindre? d'une surface? du disque?
2. Quel est le temps de latence maximal?
3. Quel est le temps de latence moyen?
4. Combien de temps faut-il pour transmettre le contenu d'une piste et combien de rotations peut effectuer le disque pendant ce temps?

| Caractéristique | Valeur |
|------------------------------------|------------------|
| Taille d'un secteur | 512 octets |
| Nbre de plateaux | 5 |
| Nbre de têtes | 10 |
| Nombre de secteurs | 5 335 031 400,00 |
| Nombre de cylindres | 10 000 |
| Nombre moyen de secteurs par piste | 40 000 |
| Temps de positionnement moyen | 10 ms |
| Vitesse de rotation | 7 400 rpm |
| Déplacement de piste à piste | 0,5 ms |
| Débit moyen | 100 Mo / s |

Exerice 3 :

Soit un disque de 5 000 cylindres tournant à 12 000 rotations par minute, avec un temps de déplacement entre deux pistes adjacentes égal à 0,002 ms et 500 secteurs de 512 octets par piste.

1. Quel est le temps moyen de lecture d'un bloc de 4,096 octets?
2. Calculer indépendamment le délai de latence, de positionnement et le temps de transfert. NB: on suppose que le nombre moyen de déplacements est 1/3 du nombre total de pistes.

Exercice 4 :

Soit la table de schéma suivant:

```
create table Personne (id integer not null,  
    nom      varchar(40) not null,  
    prenom   varchar(40) not null,  
    adresse  varchar(70),  
    dateNaissance date)
```

Cette table contient 300 000 enregistrements, stockés dans des blocs de taille 4 096 octets. Un enregistrement ne peut pas chevaucher deux blocs, et chaque bloc comprend un entête de 200 octets. On ignore l'entête des enregistrements.

1. Donner la taille maximale et la taille minimale d'un enregistrement. On suppose par la suite que tous les enregistrements ont une taille maximale.
2. Quel est le nombre maximal d'enregistrements par bloc?
3. Quelle est la taille du fichier?
4. Quel est le temps moyen de recherche d'un enregistrement si les blocs sont distribués au hasard sur le disque.
5. On suppose que le fichier est trié sur le nom. Quel est le temps d'une recherche dichotomique pour chercher une personne avec un nom donné?

Exercice 5 :

Reprenons une liste de 12 départements, à lire de gauche à droite et de bas en haut.

```
3  Allier; 36 Indre; 18 Cher; 75 Paris  
39 Jura; 9  Ariège; 81 Tarn; 11 Aude  
12 Aveyron; 25 Doubs; 73 Savoie; 55 Meuse;
```

La clé étant le numéro de département et on suppose qu'un bloc contient 5 enregistrements.

1. Proposez une fonction de hachage et le nombre d'entrées du répertoire.
2. Construisez une structure de hachage statique en prenant les enregistrements dans l'ordre de lecture de gauche à droite et de bas en haut.

Exercice 6 :

Soit un fichier de données tel que chaque bloc peut contenir 10 enregistrements. On indexe ce fichier avec un niveau d'index, et on suppose qu'un bloc d'index contient 100 entrées *[valeur, adresse]*.

Si n est le nombre d'enregistrements, donnez le nombre minimum de blocs en fonction de n pour un index dense et un index non-dense.

Exercice 7 :

Soit la liste des départements suivants, à lire de gauche à droite et de bas en haut.

```
3  Allier; 36 Indre; 18 Cher; 75 Paris  
39 Jura; 9  Ariège; 81 Tarn; 11 Aude  
12 Aveyron; 25 Doubs; 73 Savoie; 55 Meuse;  
15 Cantal; 51 Marne; 42 Loire; 40 Landes
```

14 Calvados; 30 Gard; 84 Vaucluse; 7 Ardèche

1. Construire, en prenant comme clé le numéro de département, un index dense à deux niveaux sur le fichier contenant les enregistrements dans l'ordre indiqué ci-dessus, en supposant 2 enregistrements par bloc pour les données, et 8 par bloc pour l'index.
2. Construire un index non-dense sur le fichier trié par numéro, avec les mêmes hypothèses.

Exerice 8 :

Soit les deux tables suivantes:

```
create table R (idR varchar(20) not null,  
               primary key (idR));  
  
create table S (idS int not null,  
               idR varchar(20) not null,  
               primary key (idS),  
               foreign key idR references R);
```

Indiquez, pour les ordres SQL suivants, quels index peuvent améliorer les performances ou optimiser la vérification des contraintes « primary key » et « foreign key ».

```
select * from R where idR = 'Bou'  
select * from R where idR like 'B%'  
select * from R where length(idR) = 3  
select * from R where idR like '_ou'  
insert into S values (1, 'Bou')  
select * from S where idS between 10 and 20  
delete from R where idR like 'Z%'
```